

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. A method for modifying a host user interface (UI), the method comprising:  
storing a UI for a command in a uniquely identified command item;  
obtaining from a host a uniquely identified insert location available to insert the command's stored UI into the host's UI; and  
requesting the host to insert the command's stored UI into the host UI at the uniquely identified insert location.
2. The method of Claim 1, wherein storing a UI for the command in a uniquely identified command item further comprises:  
generating a unique universal identifier (UUID) for the command; and  
storing at least one of a text, an icon, a status text, and a hotkey together with the UUID in a command item data structure.
3. The method of Claim 2, wherein storing a UI for the command in a uniquely identified command item further comprises:  
obtaining a related command from a sub-extension;  
storing an index to the related command in the command item data structure; and  
requesting the host to insert the related command's stored UI into the host UI at the uniquely identified insert location, wherein the related command's stored UI is in its own command item data structure accessible to the host via the index.
4. The method of Claim 3, wherein the related command is one of a sibling of the command and a child of the command.
5. The method of Claim 1, wherein obtaining from the host the uniquely identified insert location includes requesting from the host available insert locations and receiving from the host at least one UUID identifying each of the available insert locations.
6. A method for modifying a host user interface (UI), the method comprising:

loading at least one extension;  
uniquely identifying for the at least one extension an available insert location where the extension may request to insert commands into a host UI;  
obtaining from the at least one extension at least one uniquely identified command to insert at the available insert location; and  
integrating the command in the available insert location in accordance with the extension's load order.

7. The method of Claim 1, wherein the uniquely identified command specifies a UI to be used when integrating the command into the host UI.

8. The method of Claim 1, wherein the specified UI includes at least one of display text, a display icon, a status display text, a help text, an accelerator key, and an index to any related commands, and wherein integrating the command in the available insert location of the host UI is performed in accordance with the specified command UI.

9. The method of Claim 1, further comprising remapping the insert locations into a different portion of the host UI than in prior versions of the host UI, and integrating the command into the host UI in the remapped insert locations.

10. The method of Claim 9, the method further comprising:  
retaining a specified command in an a portion of the host UI associated with the original insert location even after remapping the insert location to the different portion of the host UI.

11. A method of communicating between a host and an extension, the method comprising:

an extension storing a UI for a command in a command item in preparation for communicating with a host;

a host issuing a call to the extension to return a number of command items to be inserted into an insert location;

the extension returning the number of command items to be inserted into the insert location;

the host issuing a call to an extension to return the command items to be inserted into the insert location;

the extension returning the command items to be inserted into the insert location; and

the host integrating the command for each of the returned command items into a host UI in accordance with each command item's stored UI.

12. The method of Claim 11, further comprising:

the host issuing a call to the extension to return a status of the command when the integrated command has become visible in the host UI; and

the extension issuing a call to the host returning the status of the command; and

the host modifying a display of the integrated command in accordance with the returned status of the command.

13. The method of Claim 12, wherein the integrated command has become visible in the host UI when the integrated command is displayed in a menu item in the host UI, and the menu item is being dropped down in response to a user action.

14. The method of Claim 11, further comprising:

the host issuing a call to the extension to execute the command when the integrated command has been invoked in the host UI; and

the extension issuing a call to the host returning the result of the executed command; and

the host modifying a display of the integrated command in accordance with the returned result of the executed command.

15. A system to safely modify a host user interface (UI) with extensions, the system comprising:

a memory in which to store a command item representing a UI for a command, and an insert location available to integrate the command into a host UI;

a processing unit operable to return the command item in response to a request for commands to integrate into the host UI at the insert location; and

a display device to display the command integrated into the host UI at the insert location in accordance with command item UI.

16. The system of Claim 15, wherein the processing unit is operable to integrate multiple commands into the host UI at the insert location in accordance with a priority of an extension that generated the commands.

17. The system of Claim 16, wherein the processing unit is operable to determine the priority of the extension that generated the command based on an order in which the extension was loaded into the system.

18. The system of Claim 17, wherein the memory in which to store the command item and insert location is to further store a universally unique identifier (UUID) for each of the command item and insert location to uniquely identify the command item and insert location from among a plurality of command items and insert locations.

19. The system of Claim 17, wherein the processing unit is operable to request a command status from the extension that generated the command by the command's UUID.

20. The system of Claim 17, wherein the processing unit is operable to request from the extension commands to be inserted at an insert location of a host UI by the insert location's UUID.

21. A computer-accessible medium having components for safely modifying a host user interface with an extension user interface, the medium comprising:

a user interface (UI) resource having a command item data structure in which to store a command UI, and an insert location data structure to store an available insert location in a host UI; and

a host interface to expose the available insert locations to extensions that have commands to insert into the host UI, and to receive a count of command items and command items representing the extension's commands;

an extension interface to receive the available insert locations from the host and to provide the count of command items and command items representing the extension's commands; and

a host process to integrate the extension's command into the host interface in accordance with the UI resource.

22. The computer-accessible medium of Claim 21, wherein the command item data structure includes a universal unique identifier (UUID) with which to uniquely identify the command, and the command UI includes at least one of a text, a icon, a status text, and help text, each of which is used to display the command in the host UI.

23. The computer-accessible medium of Claim 21, wherein the insert location data structure includes a universal unique identifier (UUID) with which to uniquely identify the insert location, wherein the insert location represents an actual location within the host UI.

24. The computer-accessible medium of Claim 21, further comprising:

a declarative list having a command item data structure in which to store an extension command UI, and an insert location data structure to store an available insert location in a extension UI;

an extension interface to expose available insert locations to sub-extensions that have commands to insert into an extension UI, and to receive a count of command items and command items representing the sub-extension's commands;

a sub-extension interface to receive the available insert locations from the extension and to provide the count of command items and command items representing the sub-extension's commands; and

an extension process to integrate the sub-extension's command into the extension UI prior to providing to the host the count of command items and command items representing the extension's commands.